REMARKS/ARGUMENTS

Reconsideration of this application in view of the above amendments and the remarks below is respectfully requested. By this amendment, Claims 1, 18 and 21 have been amended. Withdrawn Claims 11-14 have been canceled. No claims have been added. Hence, Claims 1-10 and 15-21 are pending in the application.

THE REJECTIONS BASED ON THE PRIOR ART

Claims 1-5 and 8-10 are rejected under 35 U.S.C. 103(a) as allegedly unpatentable over Burroughs et al., U.S. Patent No. 6,076,090 (hereinafter *Burroughs*) in view of Tanaka et al. U.S. Patent No. 5,375,237 (hereinafter *Tanaka*), and further in view of Exertier, U.S. Patent No. 5,832,498 (hereinafter *Exertier*). Applicant submits that Claims 1-5 and 8-10, as amended, are patentable over *Burroughs* in view of *Tanaka* in further view of *Exertier*. Each of the pending claims is discussed herein.

Claim 1

Independent method claim 1 recites:

receiving a request to execute a query;
in response to receiving the request to execute the query, before compiling the query
    for query execution, performing:
    determining that a collection of data elements to be returned by the query
        corresponds to a first data structure containing data fields, wherein **the
        data fields are not specified by a data type definition within a type
        dictionary of the database system**;
    obtaining attribute values that respectively describe the data fields within the
        first data structure; and
    recording, within the type dictionary, a first data type definition that specifies the
        data fields described by the attribute values;
    and
**removing the first data type definition from the type dictionary when one of the
    following events occurs**: a) execution of the query is complete, b) a compilation
    of the query is deleted from system memory, or c) a process identifies a flag, in
    the first data type definition, that the first data type definition is a query duration
    type. (emphasis added).

The method of Claim 1 provides an advantageous way for a database system to dynamically obtain and record data type information (or attribute values) that specifies data fields in data elements to be returned by a query. According to Claim 1, data elements to be returned by a requested query may be determined. The determined data elements may correspond to a data structure that contains data fields not specified by a data type definition within a type dictionary of the database system. Hence, attribute values that respectively describe the data fields (not specified in the type dictionary) may be obtained. Accordingly, a data type definition by which the data fields are specified may then be recorded within the type dictionary of the database system.

Thus, under one embodiment of the present invention, if the data elements to be returned by the requested query are, for example, of an opaque data type such as a binary large object (blob) that contains data fields not specified by any data type definition within a type dictionary of a database system, a data type definition that specifies the data fields in the opaque data type may still be recorded in the type dictionary. As a result, the query result to be returned to a requestor may contain a better description about the data fields contained in the opaque object.

Such a method is neither disclosed nor suggested by the cited references. For example, *Burroughs* discloses a method of translating Java language types into database types and then storing the translated Java language types in corresponding columns of a relational table to persist an object (col. 3 lines 62-64).

As disclosed by *Burroughs*, fields within the object are examined using Java Reflection methods in a storing (which is referred to as "persisting" by the reference) transaction (col. 3 lines 57-61). The primitive language fields (such as integer, character, long, float, string, etc.) found in the object are stored in columns of corresponding types in a relational table (FIG. 8

and its accompanying text). Similarly, a complex field in the object, such as Hashtable and Vector types in Java, may be stored in a column of binary format in the form of a (bit) stream (col. 4 lines 20-23). A schema map object is used to store the primitive language fields and bytes for Java Hashtable or Vector using ODBC calls.

According to *Burroughs*, neither retrieval (which is referred to as "restoring" by the reference) nor deletion involves obtaining attribute values for the fields of the Java object using the Java Reflection methods. For example, to retrieve, only an object identifier is needed to retrieve row data stored in the relational database tables.

Thus, *Burroughs* only discloses that in a storing transaction a Java class object may be dynamically examined by Java Reflection methods to obtain Java primitive types and that primitive type or Hashtable/Vector data fields in the Java class object may be stored in the relational database tables by ODBC calls using data types that are intrinsically supported by the relational database. There is no disclosure in *Burroughs* about (dynamically) recording a previously non-existent data type definition in the type dictionary of the database **at the time of responding to (i.e., in response to) a request to execute a query**.

Tanaka discloses a computer system that uses a first dictionary and a second dictionary to register and to manage definition data required to execute program products. FIG. 3 of *Tanaka* describes that a command to create program product information in a first dictionary is read from terminal 19 (as opposed to receiving a request to execute a query as featured in Claim 1). Subsequently, program product registration information in the first dictionary and a second dictionary may be created in step 65 and step 55. In the same vein, FIG. 8 of *Tanaka* describes that a command to reduce information in the first dictionary is read from terminal 19 (as opposed to receiving a request to execute a query as featured in Claim 1). Subsequently, the first dictionary may be reduced in step 227 of FIG. 8. Notably, *Tanaka* further discloses using

DDL to reduce the first dictionary, clearly indicating that, to reduce definition information from the first dictionary, *Tanaka* simply drops tables in response to a reducing command read from the terminal (as opposed to removing a data definition entry from a type dictionary).

As in the case of *Burroughs*, nothing in *Tanaka* discloses the creation of the program product registration information in the first dictionary or in the second dictionary is in response to receiving a request to execute a query. Furthermore, nothing in *Tanaka* discloses that the deletion, or what is referred to as "reducing" by the reference, of program product registration information in the first dictionary or in the second dictionary occurs at "when one of the following events occurs: a) execution of **the query** is complete, b) a compilation of the query is deleted from system memory, or c) a process identifies a flag, in the first data type definition, that the first data type definition is a query duration type", as featured in Claim 1.

Type Dictionary

Methods disclosed by the references as discussed above are quite different from that claimed in Claim 1.

The Office Action ("determining that a collection …") apparently correlates data fields in a Java class object of *Burrough* to the data fields of Claim 1, and correlates a (relational) table of *Burrough* to the data type definition of Claim 1. However, this analogy fails, because there is no disclosure in *Burroughs* that such a relational table is in the type dictionary of the database system, as required by Claim 1.

As clearly disclosed by *Burroughs*, Java class fields are read out by Java Reflection methods into primitive type or byte stream of Java language and henceforth stored in the database by ODBC calls (in which primitive types and byte streams are directly passed without any mapping that *Burroughs* has allegedly disclosed). There is no disclosure in *Burroughs* that

at the time of a query, these Java class fields are recorded in a type definition of the type dictionary of the database.

Burroughs at most discloses that Java class fields may be stored into native data types supported by a relational database through ODBC calls. Since the fields in the Java class object have been converted into the database data types through ODBC calls, there is simply no need, nor is it disclosed, in *Burroughs* for a type dictionary of the database system that contains a type definition for the Java class fields. The Java class fields are only acted upon by Java Reflection methods. The Java Reflection methods (or the schema map object), in turn, only determines what fields the Java class object contains, but is not responsible for mapping the Java class object fields to anything other than Java language primitive types or byte streams. The support for Java language primitive types or byte streams and for their corresponding relational database types are intrinsically supported by ODBC calls, without a need for ODBC to record any previously non-existent data definition in the type dictionary of the database.

Removing the First Data Type Definition

Claim 1 features a number of inter-related steps:

- A request to execute a query is received.

- In response to receiving the request to execute the query, before compiling the query for query execution, a number of steps are performed.

    o It is determined that a collection of data elements to be returned by the query corresponds to a first data structure containing data fields.

    o Attribute values that respectively describe the data fields within the first data structure are obtained.

    o A first data type definition that specifies the data fields described by the attribute values is recorded within the type dictionary.

- The first data type definition is removed from the type dictionary when one of the following events occurs: a) execution of the query is complete, b) a compilation of the query is deleted from system memory, or c) a process identifies a flag, in the first data type definition, that the first data type definition is a query duration type.

Conceding that *Burrough* may fail to expressly disclose the removing step of Claim 1, the Office Action instead contends that *Tanaka* at col. 17 lines 30-64 discloses "removing the first data type definition from the type dictionary" and that *Exertier* at col. 5 line 33 through col. 6 line 27 discloses removing when an event that execution of the query is complete occurs.

Respectfully, this argument is a mischaracterization of both Claim 1 and the references. In sharp contrast to the cited references, Claim 1 features that a first data type definition that is recorded into a type dictionary after (i.e., in response to) receiving a request to execute a query and before compiling the query for query execution, and that this first data type thus created is removed when a specified event of a) through c), as enumerated above, occurs.

Not only *Burroughs* fails to expressly disclose the requisite features of Claim 1, but also none of the references cited by the Office Action, taken individually or in combination, discloses that the first data type definition is generated in response to receiving a request to execute a query or that the first data type definition thus created is removed upon one of the specified events.

For example, *Tanaka* only discloses that computer product registration information may be reduced when a **reducing command is read** from a terminal. Contrary to the assertion of the Office Action that *Tanaka* in combination with *Exertier* discloses removing when execution of the query as featured in Claim 1, there is no disclosure in *Tanaka* or *Exertier* of a first data type that is recorded in response to **receiving a request to execute a query** and removed upon such an event as that execution of the query is complete. The Office Action inappropriately

ignores the positively recited features of Claim 1 and reads non-existent features into reference without factual support as provided by the disclosure of the references.

Applicant respectfully submits that reducing computer product registration information after a reducing command is read from a terminal is not the same as removing a first data type, which is recorded in response to receiving a request to execute **a query**, upon that execution of **the query** is complete occurs, or upon another event that is specified in Claim 1.

In fact, *Exertier* only discloses that memory space occupied by an object may be cleared upon destruction of the object. There is no disclosure in *Exertier* of a query or the memory space that is recorded in response to receiving a request to execute the query and removed upon a specified event as featured in Claim 1, even if the memory space of *Exertier* is hypothetically analogous to the first data type of Claim 1. Clearing memory space occupied by an object upon destruction of the object is not the same as removing a first data type, which is recorded in response to receiving a request to execute **a query**, upon that execution of **the query** is complete occurs, or upon another event that is specified in Claim 1.

For the reasons given above, Applicant submits that Claim 1, as amended, is patentable over the references.

Claims 18 and 21

Claims 18 and 21 are database system and computer-readable medium claims, which are analogous to the method Claim 1. Applicant submits that Claims 18 and 21 are patentable over the references for at least the same reasons as those given above in connection with Claim 1.

Claims 2-5, 8-10, 19 and 20

Claims 2-5, 8-10, 19, and 20 depend from, and hence, incorporate all of the limitations of Claim 1 or 18. These claims also recite further limitations that render them patentable over

the references. Applicant submits that these claims are patentable over the references for at least the reasons given above in connection with Claim 1.

Official Notices

Claims 6, 7 and 15-17 are rejected under 35 U.S.C. 103(a) as allegedly unpatentable over *Burroughs* in view of *Tanaka* and *Exertier*, and in further view of Official Notice. The rejection is respectfully traversed.

In rejecting these claims, the Office Action apparently follows a three-step method: 1) first concluding that Applicant has inadequately traversed the Official Notices without any reasoning why traversing is inadequate, 2) citing MPEP 2144.03 without providing any reasoning as how the cited section is applied in the instant case, and 3) concluding yet again without any reasoning that, because of Applicant's inadequate traversal, it is noted that the rejections of claims 6, 7 and 15-17 have been sustained and are to be taken as admitted prior art.

Respectfully, Applicant has traversed the Official Notices properly in the responses to the Office Actions. For example, with respect to Claim 6, Applicant at least states in previous responses as follows:

> The Office Action takes Official Notice as follows:
>
> [I]t would have been obvious to one of ordinary skill in the art at the time the invention was claimed that a binary large object (i.e., "blob") be returned, wherein a blob is a collection of binary data stored as a single entity in a database management system. Therefore, where it would have been obvious to one of ordinary skill in the art at the time the invention was claimed that an attribute value describe a blob, it would have been obvious to one of ordinary skill in the art that said attribute value(s) be returned accordingly.
>
> However, Claim 6 recites

The method of claim 1 further comprising determining whether any of the attribute values describes a data field having a plurality of component data fields.

Thus, **even if the Official Notice were correct, the Notice still fails to disclose a step of determining whether any of the attribute values describes a data field *having a plurality of component data*, as recited in Claim 6. (Emphasis added)**

As emphasized, Applicant has pointed out that what has been taken as the Official Notice (i.e., "that an attribute value describe a blob, it would have been obvious to one of ordinary skill in the art that said attribute value(s) be returned accordingly", which the Official Notice allegedly asserts as true) is different from what has been claimed in Claim 6 (i.e., "determining whether any of the attribute values describes a data field having a plurality of component data fields"). In other words, Applicant has properly traversed the Official Notice by particularly pointing out that the Official Notice is not on point as far as Claim 6 is concerned. The same discussion above applies to the other Official Notices asserted by the Office Action against other claims such as Claims 7 and 15-17.

In addition, Applicant has also specifically requested that support for each of the Official Notices be provided by Examiner. Again, Applicant has properly traversed each of the Official Notices by particularly pointing out that the alleged facts in the Official Notices have yet to meet the standard of being properly taken as such, until Examiner satisfies the burden of proof in establishing each of the Official Notices.

For at least these reasons, the conclusions established by the Office Action with regards to the Official Notice are not sustainable. For ease of reference, Applicant's previous response regarding the Official Notices is substantially reproduced hereinafter.

Preliminary Matter Regarding the Official Notices

The present Office Action again provides no supporting evidence for the Official

Notices previously improperly taken. Citing MPEP 2144.03, the Office Action asserts that

Applicant has inadequately traversed the Official Notice and that therefore the rejections of

Claims 6, 7 and 15-17 are to be taken as admitted prior art.

The Office Action appears to be confused about who legally has the burden of proof.

The cited MPEP guideline provides:

> To adequately traverse such a finding, an applicant must specifically point out the supposed errors in the examiner's action, which would include stating why the noticed fact is not considered to be common knowledge or well-known in the art. See 37 CFR 1.111(b). See also *Chevenard,* 139 F.2d at 713, 60 USPQ at 241 (**"[I]n the absence of any demand by appellant for the examiner to produce authority for his statement, we will not consider this contention."**). A general allegation that the claims define a patentable invention without any reference to the examiner's assertion of official notice would be inadequate. If applicant adequately traverses the examiner's assertion of official notice, the examiner must provide documentary evidence in the next Office action if the rejection is to be maintained. (Emphasis added)

As cited by this MPEP section, the case law is clear as to whom the burden of

proof should be carried. That is, once an applicant demands for the examiner to produce

authority for his statement, the alleged common knowledge or well-known facts asserted

by an examiner will not be established as such, unless the examiner provides

documentary evidence in the next office Action.

Here, Applicant has specifically quoted each of the statements containing the

improperly taken Official Notices in the previous response and explicitly requested that

support be provided as to why the alleged statements are well known in the art or capable

of instant and unquestionable demonstration   Therefore, Applicant has made demands

for the examiner to produce authority for his statement(s). The present Office Action

does not provide any documentary evidence to establish the alleged statements as asserted

before. Thus, under the case law, the burden of proof remains with the Examiner.

Contrary to the assertion in the Office Action, merely citing the MPEP guideline does not

equate to providing documentary evidence to support the alleged Official Notices.

Therefore, removal of the rejections as to Claims 6, 7 and 15-17 on the grounds of

improperly taken Official Notices is respectfully requested.

Claims 6 and 7

The Office Action takes Official Notice as follows:

[I]t would have been obvious to one of ordinary skill in the art at the time the
invention was claimed that a binary large object (i.e., "blob") be returned, wherein
a blob is a collection of binary data stored as a single entity in a database
management system. Therefore, where it would have been obvious to one of
ordinary skill in the art at the time the invention was claimed that an attribute
value describe a blob, it would have been obvious to one of ordinary skill in the
art that said attribute value(s) be returned accordingly.

However, Claim 6 recites

The method of claim 1 further comprising determining whether any of the
attribute values describes a data field having a plurality of component data
fields.

Thus, even if the Official Notice were correct, the Notice still fails to disclose a step of

determining whether any of the attribute values describes a data field *having a plurality of*

component data, as recited in Claim 6.

Likewise, Claim 7 recites

The method of claim 6 further comprising obtaining attribute values that describe
the plurality of component data fields.

Thus, even if the Official Notice were correct, the Notice still fails to disclose an

obtaining step for the attribute values describes a data field *having a plurality of component*

data, as recited in Claim 7.

Applicant respectfully submits that the Official Notice has been improperly taken. According to MPEP, "Official notice unsupported by documentary evidence should only be taken by the examiner where the facts asserted to be well-known, or to be common knowledge in the art are capable of instant and unquestionable demonstration as being well-known" (2144.03.A). Applicant respectfully requests that support be provided as to why the alleged fact taken under the Official Notice is well known in the art or capable of instant and unquestionable demonstration.

Furthermore, Claims 6 and 7 depend from, and hence, incorporate all of the limitations of Claim 1. For the reasons set forth above, Applicant respectfully submits that Claims 6 and 7 are patentable over the references in view of the Official Notice.

Claim 15

The Official Action also takes Official Notice as follows:

[I]t would have been obvious to one of ordinary skill in the art at the time the invention was claimed that when a function such as a SQL statement is executed, a collection of aggregate data values is returned.

However, Claim 15 recites

The method of claim 1 wherein receiving a request to execute a query comprises receiving a request to execute a function that returns a collection of aggregate data values.

Thus, even if the Official Notice were correct, the Notice still fails to disclose receiving a request to execute a function that returns a collection of aggregate data values, as recited in Claim 15.

Applicant respectfully submits that the Official Notice has been improperly taken. According to MPEP, "Official notice unsupported by documentary evidence should only be taken by the examiner where the facts asserted to be well-known, or to be common knowledge

in the art are capable of instant and unquestionable demonstration as being well-known" (2144.03.A). Applicant respectfully requests that support be provided as to why the alleged fact taken under the Official Notice is well known in the art or capable of instant and unquestionable demonstration.

Furthermore, Claim 15 depends from, and hence, incorporates all of the limitations of Claim 1. For the reasons set forth above, Applicant respectfully submits that Claim 15 is patentable over the references in view of the Official Notice.

Claim 16

The Official Action further takes Official Notice as follows:

[I]t would have been obvious to one of ordinary skill in the art at the time the invention was claimed that queries commonly indicate the type of value (e.g., an integer, text, or string) to be returned by a query.

However, Claim 16 recites

The method of claim 1 wherein determining that a collection of data elements to be returned by the query corresponds to a first data structure containing data fields not defined within a type dictionary of the database system comprises determining that a predetermined return type is indicated by the query.

Thus, even if the Official Notice were correct, the Notice still fails to disclose determining that a predetermined return type is indicated by the query. In fact, as commonly known, results from a query do not necessarily indicate any type information, for example, where the results are displayed on a monitor in ASCII.

Applicant respectfully submits that the Official Notice has been improperly taken. According to MPEP, "Official notice unsupported by documentary evidence should only be taken by the examiner where the facts asserted to be well-known, or to be common knowledge in the art are capable of instant and unquestionable demonstration as being well-known"

(2144.03.A). Applicant respectfully requests that support be provided as to why the alleged fact taken under the Official Notice is well known in the art or capable of instant and unquestionable demonstration.

Furthermore, Claim 16 depends from, and hence, incorporates all of the limitations of Claim 1. For the reasons set forth above, Applicant respectfully submits that Claim 16 is patentable over the references in view of the Official Notice.

Claim 17

Claim 17 depends from, and hence, incorporates all of the limitations of Claim 1. Claim 17 also recites further limitations that render it patentable over the references. Applicant submits that Claim 17 patentable over the references in view of the Official Notice for at least the reasons given above in connection with Claim 1.

CONCLUSION

For the reasons set forth above, it is respectfully submitted that all of the pending claims are now in condition for allowance. Therefore, the issuance of a formal Notice of Allowance is believed next in order, and that action is most earnestly solicited.

The Examiner is respectfully requested to contact the undersigned by telephone if it is believed that such contact would further the examination of the present application.

Please charge any shortages or credit any overages to Deposit Account No. 50-1302.

Respectfully submitted,

HICKMAN PALERMO TRUONG & BECKER LLP

/ZhichongGu#56543/

Zhichong Gu
Reg. No. 56,543

Date: February 25, 2008

2055 Gateway Place, Suite 550
San Jose, CA 95110
Direct: (408) 414-1236
Facsimile: (408) 414-1076